FB.DM\_BOXED\_PRIMITIVE\_FOR\_COMPARE

Упакованный примитив создается только для вызова метода CompareTo(). Более эффективно использовать метод статического сравнения (для двойных значений и чисел с плавающей запятой, начиная с Java 1.4, для других примитивных типов, начиная с Java 7), который работает непосредственно с примитивами

Таблица №1 – Основные характеристики детектора FB.DM\_BOXED\_PRIMITIVE\_FOR\_COMPARE

|  |  |
| --- | --- |
| Категория критичности | Minor |
| Надежность | Unknown |
| Источник | SpotBugs |
| Тип ошибки | Performance Issue |
| CWE | --- |
| CVE | --- |
| Возможные последствия | * Избыточное создание объектов * Снижение производительности * Увеличение нагрузки на GC |

Таблица №2 – Примеры срабатывания детектора и устранения уязвимости

|  |  |
| --- | --- |
| Потенциально небезопасное | Корректная конструкция |
| public static void incorrectTest() {  // Некорректно: использование boxed типов для сравнения  Integer a = 100;  Integer b = 100;    if (a.compareTo(b) == 0) { // Избыточное boxed сравнение  System.out.println("Values are equal (boxed)");  }  } | public static void correctTest() {  // Корректно: сравнение примитивов  int a = 100;  int b = 100;    if (a == b) { // Простое сравнение примитивов  System.out.println("Values are equal (primitive)");  }    // Или для обязательного boxed сравнения - статический метод  Integer x = 200;  Integer y = 200;  if (Integer.compare(x, y) == 0) { // Без создания промежуточных объектов  System.out.println("Boxed values compared efficiently");  }  } |